Thomas Studer
Relationale Datenbanken:
Von den theoretischen Grundlagen
zu Anwendungen mit PostgreSQL
Springer, 2016
ISBN 978-3-662-46570-7

Datenbanken

—

Normalformen

Thomas Studer

Institut für Informatik
Universität Bern

Bibliotheksverwaltung

$\mathcal{B}$ modelliert den Buchbestand der Bibliothek, $\mathcal{A}$ modelliert die Ausleihen:

$$\mathcal{B} := (\underline{\texttt{BId}}, \texttt{ISBN}, \texttt{Titel}, \texttt{Autor})$$
$$\mathcal{A} := (\underline{\texttt{BId}}, \texttt{Name}, \texttt{Adresse}, \texttt{Datum})$$

Das Attribut `BId` ist eine eindeutige Id für die Bücher, welche zum Bestand der Bibliothek gehören.
Für jedes dieser Bücher wird die ISBN, der Titel und der Autor abgespeichert.
Bei einer Ausleihe, wird die Id des ausgeliehenen Buches, der Name und die Adresse der Benutzers, welcher das Buch ausleiht, sowie das Ausleihdatum abgespeichert.
Wir nehmen hier an, dass ein Benutzer eindeutig durch seinen Namen identifiziert ist.
Ausserdem sei `BId` im Schema $\mathcal{A}$ ein Fremdschlüssel auf das Schema $\mathcal{B}$.

Änderungsanomalie

Nehmen wir an, eine Benutzeradresse soll geändert werden.

Falls dieser Benutzer mehrere Bücher ausgeliehen hat, so gibt es mehrere Einträge mit diesem Benutzer in der $\mathcal{A}$-Relation. In jedem dieser Einträge muss nun die Adresse aktualisiert werden.

Das heisst, obwohl nur die Adresse *eines* Benutzers ändert, müssen *mehrere* Tupel aktualisiert werden.

Einfügeanomalie

Ein neuer Benutzer kann nur erfasst werden, falls er auch zugleich ein Buch ausleiht.

Wenn sich jemand neu anmeldet ohne ein Buch auszuleihen, dann können seine Daten (Name und Adresse) nicht in der Datenbank eingetragen werden.

Löschanomalie

Wenn ein Benutzer alle ausgeliehenen Bücher zurückbringt, dann werden alle Daten über diesen Benutzer in der $\mathcal{A}$-Relation gelöscht.

Es sind damit keine Informationen mehr über ihn gespeichert. Somit müssen bei einer neuen Ausleihe alle Benutzerdaten wieder neu erfasst werden.

Analyse

Der Grund für diese Anomalien ist, dass durch *ein* Schema *mehrere* Konzepte modelliert werden.

Tatsächlich werden im Schema $\mathcal{A}$ sowohl Informationen über die Ausleihen als auch über die Benutzer abgespeichert.

Auch das Schema $\mathcal{B}$ enthält Information über zwei Konzepte: nämlich den aktuellen Buchbestand und die Buchausgaben.

Funktionale Abhängigkeit

Es seien $A_1, \ldots, A_n$ Attribute. Eine *funktionale Abhängigkeit* (functional dependency) auf einer Attributmenge $\{A_1, \ldots, A_n\}$ ist gegeben durch

$$X \to Y \quad ,$$

wobei $X, Y \subseteq \{A_1, \ldots, A_n\}$.

Eine funktionale Abhängigkeit $X \to Y$ heisst *trivial*, falls $Y \subseteq X$ gilt.

R *erfüllt* $X \to Y$

Sei R eine Relation über den Attributen $A_1, \dots, A_n$. Ferner sei $X \subseteq \{A_1, \dots, A_n\}$. Für $s, t \in R$ schreiben wir

$$s[X] = t[X] \ ,$$

falls $s[A_i] = t[A_i]$ für alle $A_i \in X$ gilt.

Gegeben sei eine Relation R eines Schemas $\mathcal{S}$. Ferner sei $X \to Y$ eine funktionale Abhängigkeit auf $\mathcal{S}$. Die Relation R *erfüllt* $X \to Y$, falls für alle Tupel $s, t \in R$ gilt

$$s[X] = t[X] \quad \Longrightarrow \quad s[Y] = t[Y] \ .$$

Gegeben seien ein Schema $\mathcal{S}$ und eine *triviale* funktionale Abhängigkeit $X \to Y$ auf $\mathcal{S}$. Offensichtlich erfüllt jede Relation auf $\mathcal{S}$ die funktionale Abhängigkeit $X \to Y$.

Annahme

Annahme

Im Teil über Normalformen enthalten alle vorkommenden Datenbanken enthalten keine `Null` Werte.

Beispiel

Ausleihen

<u>Bid</u>	Name	Adresse	Datum
11	Eva	Thun	20140506
5	Eva	Thun	20140804
4	Tom	Bern	20140301

Erfüllt

$$\{\mathtt{Bid}\} \to \{\mathtt{Name}, \mathtt{Adresse}, \mathtt{Datum}\}$$

und

$$\{\mathtt{Name}\} \to \{\mathtt{Adresse}\} \ .$$

Verletzt

$$\{\mathtt{Name}\} \to \{\mathtt{Datum}\} \ .$$

Unique Constraints

Gegeben sei DB-Schema $\mathcal{S} = (A_1, \ldots, A_n)$. Ein Unique constraint

$$U = (A_{i_1}, \ldots, A_{i_m})$$

kann durch folgende funktionale Abhängigkeit ausgedrückt werden:

$$\{A_{i_1}, \ldots, A_{i_m}\} \ \rightarrow \ \{A_1, \ldots, A_n\} \ .$$

Abkürzungen

Es sei $\mathcal{S} = (A_1, \ldots, A_n)$ ein Schema und $X, Y, Z \subseteq \{A_1, \ldots, A_n\}$.

1. Wir verwenden YZ für $Y \cup Z$. Somit steht beispielsweise

$$X \to YZ \quad \text{für} \quad X \to Y \cup Z \ .$$

2. Wir schreiben $\mathcal{S}$ für $\{A_1, \ldots, A_n\}$. Damit betrachten wir $\mathcal{S}$ als ungeordnete Menge und

$$X \to \mathcal{S} \quad \text{steht für} \quad X \to \{A_1, \ldots, A_n\} \ .$$

3. Wir verwenden A_i für die einelementige Menge $\{A_i\}$. Somit steht beispielsweise

$$X \to A_i \quad \text{für} \quad X \to \{A_i\} \ .$$

Damit können wir auch

$$X \to A_i A_j \quad \text{für} \quad X \to \{A_i, A_j\}$$

schreiben.

Logische Folgerung

Gegeben sei ein Schema $\mathcal{S} = (A_1, \ldots, A_n)$. Ferner seien eine Menge F von funktionalen Abhängigkeiten über $\mathcal{S}$ sowie eine weitere funktionale Abhängigkeit $X \to Y$ über $\mathcal{S}$ gegeben.
Wir sagen, $X \to Y$ *folgt logisch aus* F, in Zeichen

$$F \models X \to Y \quad ,$$

falls jede Instanz R von $\mathcal{S}$, welche alle Abhängigkeiten in F erfüllt, auch $X \to Y$ erfüllt.

Beispiel: Gegeben sei die Menge $F = \{W \to X,\ W \to Y,\ XY \to Z\}$ von funktionalen Abhängigkeiten. Dann gilt unter anderem:

1. $F \models W \to X$,
2. $F \models W \to XY$,
3. $F \models W \to Z$.

Hülle F^+

Ist F eine Menge von funktionalen Abhängigkeiten, so wird die *Hülle* F^+ von F definiert durch

$$F^+ := \{X \to Y \mid F \models X \to Y\} \ .$$

Schlüssel

Sei $\mathcal{S} = (A_1, \ldots, A_n)$ ein Schema mit einer Menge von funktionalen Abhängigkeiten F. Falls F eine nicht-triviale funktionale Abhängigkeit enthält, dann gibt es eine echte Teilmenge X von $\{A_1, \ldots, A_n\}$, so dass

$$X \to \{A_1, \ldots, A_n\} \in F^+ \quad .$$

Gegeben sei ein Schema $\mathcal{S} = (A_1, \ldots, A_n)$ mit einer Menge von funktionalen Abhängigkeiten F. Eine Teilmenge

$$X \subseteq \{A_1, A_2, \ldots, A_n\}$$

heisst *Superschlüssel* für $\mathcal{S}$ bezüglich F, falls gilt:

$$X \to \mathcal{S} \ \in \ F^+ \quad .$$

Ein Superschlüssel für $\mathcal{S}$ heisst *Schlüssel* für $\mathcal{S}$ bezüglich F, falls zusätzlich gilt:

es gibt keine *echte* Teilmenge $Y \subsetneq X$ mit $Y \to \mathcal{S} \ \in \ F^+$.

Beispiel

Wir betrachten Postleitzahlen und treffen folgende vereinfachenden Annahmen:

1. Jede Stadt ist eindeutig durch ihre Postleitzahl bestimmt. D.h. es gibt keine zwei Städte mit derselben Postleitzahl.
2. Jede Postleitzahl ist eindeutig durch Stadt und Strasse bestimmt. D.h. die Postleitzahl ändert sich innerhalb einer Strasse nicht.

Entsprechend wählen wir Attribute `Stadt`, `Str` und `PLZ` sowie das Schema

$$\mathcal{S} = (\mathtt{Stadt}, \mathtt{Str}, \mathtt{PLZ})$$

mit der dazugehörigen Menge

$$\big\{ \{\mathtt{Stadt}, \mathtt{Str}\} \to \{\mathtt{PLZ}\}, \quad \{\mathtt{PLZ}\} \to \{\mathtt{Stadt}\} \big\}$$

von funktionalen Abhängigkeiten. Mit der obigen Definition folgt sofort, dass die Attributmengen $\{\mathtt{Stadt}, \mathtt{Str}\}$ und $\{\mathtt{Str}, \mathtt{PLZ}\}$ Schlüssel von $\mathcal{S}$ sind.

Zerlegung

Gegeben seien die Attribute $A_1, A_2, \ldots, A_n$ sowie das Schema

$$\mathcal{S} \;:=\; (A_1, \ldots, A_n).$$

Eine *Zerlegung* von $\mathcal{S}$ ist eine Menge von Relationenschemata

$$\{\,(A_{11}, A_{12}, \ldots, A_{1m_1}), \ldots, (A_{k1}, A_{k2}, \ldots, A_{km_k})\,\}\ ,$$

so dass gilt

$$\{A_{11}, A_{12}, \ldots, A_{1m_1}\} \cup \cdots \cup \{A_{k1}, A_{k2}, \ldots, A_{km_k}\} \;=\; \{A_1, A_2, \ldots, A_n\}\ .$$

Notation

Sei $\mathcal{S} = (A_1, \dots, A_n)$. Zur Vereinfachung der Notation schreiben wir

$$\pi_{\mathcal{S}}(R) \quad \text{anstelle von} \quad \pi_{A_1,\dots,A_n}(R) \ .$$

Wir werden die Gleichheit von Relationen bezüglich eines Schemas $\mathcal{S}$ verwenden. Wir setzen:

$$R \ =_{\mathcal{S}} \ T \quad :\Longleftrightarrow \quad \pi_{\mathcal{S}}(R) = \pi_{\mathcal{S}}(T) \ .$$

Verlustige Zerlegung

Sei $\mathcal{S} := (\texttt{Name}, \texttt{Marke}, \texttt{Farbe})$ mit Zerlegung

$$\mathcal{S}_1 := (\texttt{Name}, \texttt{Marke}) \text{ und } \mathcal{S}_2 := (\texttt{Marke}, \texttt{Farbe}) \ .$$

Betrachte die Instanz `Autos` von $\mathcal{S}$ und ihre Zerlegung:

`Autos`

Name	Marke	Farbe
Eva	Audi	schwarz
Tom	Audi	rot

$\pi_{\mathcal{S}_1}(\texttt{Autos})$

Name	Marke
Eva	Audi
Tom	Audi

$\pi_{\mathcal{S}_2}(\texttt{Autos})$

Marke	Farbe
Audi	schwarz
Audi	rot

Damit erhalten wir:

$\pi_{\mathcal{S}_1}(\texttt{Autos}) \bowtie \pi_{\mathcal{S}_2}(\texttt{Autos})$

Marke	Name	Farbe
Audi	Eva	schwarz
Audi	Eva	rot
Audi	Tom	schwarz
Audi	Tom	rot

Verlustfreie Zerlegung

Wir gehen aus von einem Schema $\mathcal{S}$, einer Zerlegung $\{\mathcal{S}_1, \ldots, \mathcal{S}_k\}$ von $\mathcal{S}$ sowie einer Menge F von funktionalen Abhängigkeiten über den Attributen von $\mathcal{S}$. Dann besitzt die Zerlegung $\{\mathcal{S}_1, \ldots, \mathcal{S}_k\}$ einen *verlustfreien Verbund bezüglich* F, falls für alle Instanzen R von $\mathcal{S}$, die F erfüllen, gilt, dass

$$R \ =_{\mathcal{S}} \ \pi_{\mathcal{S}_1}(R) \bowtie \ldots \bowtie \pi_{\mathcal{S}_k}(R) \ .$$

In diesem Fall sprechen wir von einer *verlustfreien Zerlegung des Schemas* $\mathcal{S}$ *bezüglich* F.

Lemma

Lemma

Gegeben sei ein Schema $\mathcal{S}$ mit einer Menge F von funktionalen Abhängigkeiten. Eine Zerlegung $\{\mathcal{S}_1, \mathcal{S}_2\}$ von $\mathcal{S}$ ist genau dann verlustfrei bezüglich F, wenn

1. $\mathcal{S}_1 \cap \mathcal{S}_2 \to \mathcal{S}_1 \in F^+$ *oder*
2. $\mathcal{S}_2 \cap \mathcal{S}_2 \to \mathcal{S}_2 \in F^+$.

$\mathcal{A} := (\texttt{BId}, \texttt{Name}, \texttt{Adresse}, \texttt{Datum})$ wird zerlegt in $\{\mathcal{A}_1, \mathcal{A}_2\}$ mit

$$\begin{aligned} \mathcal{A}_1 &:= (\texttt{BId}, \texttt{Name}, \texttt{Datum}) \\ \mathcal{A}_2 &:= (\texttt{Name}, \texttt{Adresse}) \ . \end{aligned}$$

Zu $\mathcal{A}$ gehört die Menge von funktionalen Abhängigkeiten

$$F := \big\{\{\texttt{Bid}\} \to \{\texttt{Name}, \texttt{Adresse}, \texttt{Datum}\}, \{\texttt{Name}\} \to \{\texttt{Adresse}\}\big\} \ .$$

Wir finden $\mathcal{A}_1 \cap \mathcal{A}_2 = \{\texttt{Name}\}$. Also gilt $\mathcal{A}_1 \cap \mathcal{A}_2 \to \mathcal{A}_2 \in F^+$. Es folgt, dass die Zerlegung verlustfrei ist.

Abhängigkeitserhaltende Zerlegung

Wir betrachten wieder ein Schema $\mathcal{S}$, eine Zerlegung $\{\mathcal{S}_1, \ldots, \mathcal{S}_k\}$ von $\mathcal{S}$ sowie eine Menge F von funktionalen Abhängigkeiten über den Attributen von $\mathcal{S}$.

1. Die *Projektion von F auf eine Attributmenge Z* wird definiert durch

$$\Pi_Z(F) \ := \ \{ \, X \to Y \in F^+ \mid XY \subseteq Z \, \} \ .$$

2. Ist $\mathcal{T}$ das Relationenschema $(A_1, \ldots, A_n)$, so setzen wir

$$\Pi_{\mathcal{T}}(F) \ := \ \Pi_{\{A_1, \ldots, A_n\}}(F) \ .$$

3. Die Zerlegung $\{\mathcal{S}_1, \ldots, \mathcal{S}_k\}$ heisst *abhängigkeitserhaltende Zerlegung von $\mathcal{S}$ bezüglich F*, falls gilt

$$\Big(\bigcup_{i=1}^{k} \Pi_{\mathcal{S}_i}(F) \Big)^+ \ = \ F^+ \ .$$

Beispiel

Sei

$$\begin{aligned}\mathcal{S} &= (\texttt{Stadt}, \texttt{Str}, \texttt{PLZ}),\\ \mathcal{S}_1 = (\texttt{Str}, \texttt{PLZ}), &\qquad \mathcal{S}_2 = (\texttt{Stadt}, \texttt{PLZ}).\end{aligned}$$

Wie vorher sei ausserdem F unsere Menge

$$\{\, \{\texttt{Stadt}, \texttt{Str}\} \to \texttt{PLZ}, \quad \texttt{PLZ} \to \texttt{Stadt} \,\}$$

von funktionalen Abhängigkeiten. $\{\mathcal{S}_1, \mathcal{S}_2\}$ ist eine verlustfreie Zerlegung von $\mathcal{S}$ bezüglich F. Ferner gilt:

$$\begin{aligned}\Pi_{\mathcal{S}_1}(F) &= \left\{\, X \to Y \mid XY \subseteq \{\texttt{Str}, \texttt{PLZ}\} \text{ und } \emptyset \models X \to Y \,\right\} \;,\\ \Pi_{\mathcal{S}_2}(F) &= \{\, X \to Y \mid XY \subseteq \{\texttt{Stadt}, \texttt{PLZ}\} \text{ und}\\ &\qquad\qquad \{\texttt{PLZ} \to \texttt{Stadt}\} \models X \to Y \,\} \;.\end{aligned}$$

Jedoch haben wir

$$\Pi_{\mathcal{S}_1}(F) \cup \Pi_{\mathcal{S}_2}(F) \not\models \{\texttt{Stadt}, \texttt{Str}\} \to \texttt{PLZ} \;.$$

In der Tat, folgende Instanzen `S1` von $\mathcal{S}_1$ und `S2` von $\mathcal{S}_2$ zeigen, dass F durch die Zerlegung $\{\mathcal{S}_1, \mathcal{S}_2\}$ nicht erhalten wird:

Beispiel 2

S1

Str	PLZ
Baumstr	2500
Baumstr	2502

S2

Stadt	PLZ
Biel	2500
Biel	2502

Damit folgt nämlich

S1 $\bowtie$ S2

PLZ	Str	Stadt
2500	Baumstr	Biel
2502	Baumstr	Biel

Die Relationen S1 und S2 erfüllen jeweils die projizierten funktionalen Abhängigkeiten $\Pi_{\mathcal{S}_1}(F)$ und $\Pi_{\mathcal{S}_2}(F)$. Der Verbund S1 $\bowtie$ S2 verletzt jedoch die funktionale Abhängigkeit $\{\texttt{Stadt}, \texttt{Str}\} \rightarrow \texttt{PLZ}$.

Definitionen

Gegeben sei ein Relationenschema $\mathcal{S}$ sowie eine Menge F von funktionalen Abhängigkeiten bezüglich $\mathcal{S}$.

1. Ein Attribut A von $\mathcal{S}$ heisst *prim*, falls A Teil eines Schlüssels von $\mathcal{S}$ ist; anderenfalls heisst A *nicht-prim*.
2. Es gelte $X \to Y \in F^+$ und $Y \to X \notin F^+$; ausserdem sei A ein Attribut von $\mathcal{S}$, das weder in X noch in Y vorkommt und für das $Y \to A \in F^+$ gilt. Dann sagen wir, dass A von X *transitiv bezüglich F abhängig* ist.
3. Eine funktionale Abhängigkeit $X \to Y$ mit Attributen von $\mathcal{S}$ heisst *partielle Abhängigkeit* bezüglich F, falls es eine echte Teilmenge Z von X gibt, so dass $Z \to Y \in F^+$ ist. Dann sagen wir, dass Y von X *partiell bezüglich F abhängig* ist.

Beispiele

Betrachte $\mathcal{A} := (\texttt{Bid}, \texttt{Name}, \texttt{Adresse}, \texttt{Datum})$ mit

$$F := \big\{\{\texttt{Bid}\} \rightarrow \{\texttt{Name}, \texttt{Adresse}, \texttt{Datum}\}, \{\texttt{Name}\} \rightarrow \{\texttt{Adresse}\}\big\} \ .$$

Wir haben $\quad \texttt{Bid} \rightarrow \texttt{Name} \in F^+$ und $\texttt{Name} \rightarrow \texttt{Bid} \notin F^+ \ .$
Ausserdem gilt $\quad \texttt{Name} \rightarrow \texttt{Adresse} \in F^+ \ .$
Somit ist $\texttt{Adresse}$ von $\texttt{Bid}$ transitiv bezüglich F abhängig.

Sei $\mathcal{S}_1 := (\underline{\texttt{Autor}}, \texttt{Jahrgang}, \underline{\texttt{Titel}})$ mit $F_1 := \{\texttt{Autor} \rightarrow \texttt{Jahrgang}\}$.
Also ist

$$\{\texttt{Autor}, \texttt{Titel}\} \rightarrow \{\texttt{Jahrgang}\} \in F_1^+$$

eine partielle Abhängigkeit bezüglich F ist, da $\{\texttt{Autor}\}$ eine echte Teilmenge von $\{\texttt{Autor}, \texttt{Titel}\}$ ist und gilt

$$\{\texttt{Autor}\} \rightarrow \{\texttt{Jahrgang}\} \in F_1^+ \ .$$

1NF und 2NF

Erste Normalform (1NF): $\mathcal{S}$ ist in *erster Normalform*, falls alle Attribute von $\mathcal{S}$ nur atomare Domänen haben. Dabei heisst eine Domäne *atomar*, falls ihre Elemente als nicht-unterteilbare Einheiten aufgefasst werden.

Zweite Normalform (2NF): $\mathcal{S}$ ist in *zweiter Normalform* bezüglich F, falls $\mathcal{S}$ in erster Normalform ist und für alle Attribute A von $\mathcal{S}$ mindestens eine der folgenden zwei Bedingungen erfüllt ist:

(2NF.1) A ist prim;

(2NF.2) A ist nicht von einem Schlüssel für $\mathcal{S}$ partiell bezüglich F abhängig.

3NF und BCNF

Dritte Normalform (3NF): $\mathcal{S}$ ist in *dritter Normalform* bezüglich F, falls $\mathcal{S}$ in erster Normalform ist und für alle $X \to Y$ aus F^+ mindestens eine der folgenden drei Bedingungen erfüllt ist:

(3NF.1) $Y \subseteq X$;

(3NF.2) X ist ein Superschlüssel von $\mathcal{S}$;

(3NF.3) jedes Attribut A aus $Y \setminus X$ ist prim.

Boyce–Codd Normalform (BCNF): $\mathcal{S}$ ist in *Boyce–Codd Normalform* bezüglich F, falls $\mathcal{S}$ in erster Normalform ist und für alle $X \to Y$ aus F^+ mindestens eine der folgenden zwei Bedingungen erfüllt ist:

(BCNF.1) $Y \subseteq X$;

(BCNF.2) X ist ein Superschlüssel von $\mathcal{S}$.

Nicht 1NF

Sei $\mathcal{S}_0 := (\underline{\texttt{Autor}}, \texttt{Jahrgang}, \texttt{Titelliste})$
mit $F_0 := \{\texttt{Autor} \to \texttt{Jahrgang}, \texttt{Autor} \to \texttt{Titelliste}\}$.
Wir betrachten nun folgende Instanz von $\mathcal{S}_0$:

`Werke`

<u>`Autor`</u>	`Jahrgang`	`Titelliste`
`Goethe`	`1749`	`{Götz, Faust}`
`Schiller`	`1759`	`{Tell}`

Die Domäne des Attributs `Titelliste` ist hier nicht atomar. Sie besteht aus Mengen, welche aus einzelnen Elementen zusammengesetzt sind. Somit ist dieses Schema *nicht* in 1NF.
Das Problem ist, dass nicht auf einen einzelnen Titel zugegriffen werden kann: Die Query

Wer ist der Autor von `Faust`?

kann in der relationalen Algebra nicht ausgedrückt werden.

1NF, aber nicht 2NF

Sei $\mathcal{S}_1 := (\underline{\texttt{Autor}}, \texttt{Jahrgang}, \underline{\texttt{Titel}})$
mit $F_1 := \{\texttt{Autor} \to \texttt{Jahrgang}\}$ Damit gilt

$$\{\texttt{Autor}, \texttt{Titel}\} \to \{\texttt{Autor}, \texttt{Jahrgang}, \texttt{Titel}\} \in F_1^+ \ .$$

Das heisst, $K := \{\texttt{Autor}, \texttt{Titel}\}$ ist ein Schlüssel für $\mathcal{S}_1$.

Betrachte

`Werke`

<u>`Autor`</u>	`Jahrgang`	<u>`Titel`</u>
`Goethe`	`1749`	`Götz`
`Goethe`	`1749`	`Faust`
`Schiller`	`1759`	`Tell`

Alle Attribute haben nun atomare Domänen. Somit ist dieses Schema in 1NF. Jedoch ist es *nicht* in 2NF. Es gilt nämlich:

1. `Jahrgang` ist nicht-prim und
2. `Jahrgang` ist partiell vom Schlüssel K abhängig.

Problem: die Daten des Jahrgangs sind mehrfach vorhanden.

2NF, aber nicht 3NF

Sei $\mathcal{S}_2 := (\underline{\texttt{BuchId}}, \texttt{Autor}, \texttt{Jahrgang}, \texttt{Titel})$ mit

$$F_2 := \{\{\texttt{BuchId}\} \to \{\texttt{Autor}, \texttt{Jahrgang}, \texttt{Titel}\},\ \ \{\texttt{Autor}\} \to \{\texttt{Jahrgang}\}\}$$

Betrachte

Werke

<u>BuchId</u>	Autor	Jahrgang	Titel
1	Goethe	1749	Götz
2	Goethe	1749	Faust
3	Schiller	1759	Tell
4	Schiller	1759	Tell

Das Schema $\mathcal{S}_2$ ist in 2NF: (2NF.2) ist offensichtlich erfüllt

Aber nicht in 3NF. Betrachte $\texttt{Autor} \to \texttt{Jahrgang} \in F_2^+$.

1. (3NF.1) ist nicht erfüllt: es gilt nämlich $\{\texttt{Jahrgang}\} \not\subseteq \{\texttt{Autor}\}$.
2. (3NF.2) ist nicht erfüllt: $\{\texttt{Autor}\}$ ist kein Superschlüssel von $\mathcal{S}_2$.
3. (3NF.3) ist nicht erfüllt: $\{\texttt{Jahrgang}\}$ ist nicht prim, d.h. das Attribut Jahrgang ist nicht Teil eines Schlüssels.

Keine der Bedingungen ist erfüllt und somit ist $\mathcal{S}_2$ nicht in 3NF.

Merke:

Jedes nicht-prim Attribut muss etwas aussagen über

1. den Schlüssel (1NF),
2. den ganzen Schlüssel (2NF) und
3. nur über den Schlüssel (3NF).

3NF, aber nicht BCNF

Sei $\mathcal{S}_3 = (\texttt{Stadt}, \texttt{Str}, \texttt{PLZ})$ mit
$F_3 := \{ \{\texttt{Stadt}, \texttt{Str}\} \to \{\texttt{PLZ}\}, \{\texttt{PLZ}\} \to \{\texttt{Stadt}\} \}$.
Schlüssel von $\mathcal{S}_3$ sind $\{\texttt{Stadt}, \texttt{Str}\}$ und $\{\texttt{Str}, \texttt{PLZ}\}$.

Betrachte

Verzeichnis

PLZ	Str	Stadt
2500	Baumstr	Biel
3000	Parkstr	Bern
3018	Wiesenstr	Bern
3018	Baumstr	Bern

Dieses Schema ist in 3NF (3NF.3 ist erfüllt); aber nicht BCNF.
Betrachte $\texttt{PLZ} \to \texttt{Stadt} \in F_3^+$. Wir finden:

1. (BCNF.1) ist nicht erfüllt: es gilt nämlich $\{\texttt{Stadt}\} \not\subseteq \{\texttt{PLZ}\}$.
2. (BCNF.2) ist nicht erfüllt: $\{\texttt{PLZ}\}$ ist kein Superschlüssel von $\mathcal{S}_3$.

Redundanz: Beziehung zwischen Postleitzahl 3018 und dem Ortsnamen Bern.

Frage

Wie das vorherige Beispiel zeigt, können Updates auf einem Schema in 3NF noch zu Inkonsistenzen führen.
Ist jedoch ein Schema in BCNF bezüglich einer Menge F von funktionalen Abhängigkeiten, so kann es keine Redundanzen geben, welche durch F verursacht werden.
Das heisst, in einem Schema, welches in BCNF ist, können Updates nicht zu Inkonsistenzen bezüglich funktionaler Abhängigkeiten führen.

Die Frage lautet somit:

> Gibt es zu jedem DB-Schema ein äquivalentes DB-Schema in BCNF?

Theorem

Gegeben seien ein Schema $\mathcal{S}$ und eine Menge von funktionalen Abhängigkeiten F bezüglich $\mathcal{S}$. Dann gilt:

1. Es gibt eine verlustfreie Zerlegung

$$\mathcal{Z} := \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$$

   von $\mathcal{S}$, so dass alle $\mathcal{S}_i \in \mathcal{Z}$ in BCNF bezüglich $\Pi_{\mathcal{S}_i}(F)$ sind.

2. Es gibt eine verlustfreie und abhängigkeitserhaltende Zerlegung

$$\mathcal{Z} := \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$$

   von $\mathcal{S}$, so dass alle $\mathcal{S}_i \in \mathcal{Z}$ in 3NF bezüglich $\Pi_{\mathcal{S}_i}(F)$ sind.

Beispiel (siehe: nicht 2NF)

Sei $\mathcal{S}_1 := (\underline{\texttt{Autor}}, \texttt{Jahrgang}, \underline{\texttt{Titel}})$ mit $F_1 := \{\texttt{Autor} \to \texttt{Jahrgang}\}$. Zerlegung:

$$\mathcal{S}_{1,1} := \{\texttt{Autor}, \texttt{Jahrgang}\} \text{ und } \mathcal{S}_{1,2} := \{\texttt{Autor}, \texttt{Titel}\} \enspace .$$

Verlustfrei (siehe Lemma) und abhängigkeitserhaltend. Insbesondere gilt

$$\{\texttt{Autor} \to \texttt{Jahrgang}\} \in \Pi_{\mathcal{S}_{1,1}}(F) \enspace .$$

Weiter stellen wir fest:

1. $\mathcal{S}_{1,1}$ ist in BCNF bezüglich $\Pi_{\mathcal{S}_{1,1}}(F)$,
2. $\mathcal{S}_{1,2}$ ist in BCNF bezüglich $\Pi_{\mathcal{S}_{1,2}}(F)$.

Die Relation `Werke` aus Beispiel *nicht 2NF* wird wie folgt zerlegt:

$\pi_{\mathcal{S}_{1,1}}(\texttt{Werke})$

<u>Autor</u>	Jahrgang
Goethe	1749
Schiller	1759

$\pi_{\mathcal{S}_{1,2}}(\texttt{Werke})$

<u>Autor</u>	<u>Titel</u>
Goethe	Götz
Goethe	Faust
Schiller	Tell

Beispiel: Zerlegung mit Abhängigkeitsverlust

Sei $\mathcal{S}_3 = (\texttt{Stadt}, \texttt{Str}, \texttt{PLZ})$ mit
$F_3 := \{ \{\texttt{Stadt}, \texttt{Str}\} \to \{\texttt{PLZ}\}, \{\texttt{PLZ}\} \to \{\texttt{Stadt}\} \}$ (nicht in BCNF).
Wir wählen nun die Zerlegung $\{\mathcal{S}_{3,1}, \mathcal{S}_{3,2}\}$ mit

$$\mathcal{S}_{3,1} := \{\texttt{Str}, \texttt{PLZ}\} \text{ und } \mathcal{S}_{3,2} := \{\texttt{Stadt}, \texttt{PLZ}\} \ .$$

Verlustfrei, nicht abhängigkeitserhaltend (siehe früheres Bsp).
Weiter stellen wir fest:

1. $\mathcal{S}_{3,1}$ ist in BCNF bezüglich $\Pi_{\mathcal{S}_{3,1}}(F)$,
2. $\mathcal{S}_{3,2}$ in ist BCNF bezüglich $\Pi_{\mathcal{S}_{3,2}}(F)$.

Die Relation `Verzeichnis` aus Beispiel *nicht BCNF* wird wie folgt zerlegt:

$\pi_{\mathcal{S}_{3,1}}(\texttt{Verzeichnis})$

Str	PLZ
Baumstr	2500
Parkstr	3000
Wiesenstr	3018
Baumstr	3018

$\pi_{\mathcal{S}_{3,2}}(\texttt{Verzeichnis})$

Stadt	PLZ
Biel	2500
Bern	3000
Bern	3018

Lemma: transitive Abhängigkeit und 3NF

Lemma

Gegeben seien ein Relationenschema $\mathcal{S}$ sowie eine Menge F von funktionalen Abhängigkeiten bezüglich $\mathcal{S}$. Dann ist $\mathcal{S}$ in 3NF bezüglich F genau dann, wenn es kein nicht-primes Attribut A von $\mathcal{S}$ gibt, das von einem Schlüssel für $\mathcal{S}$ transitiv bezüglich F abhängig ist.

Beweis von links nach rechts. Sei $\mathcal{S}$ in 3NF bezüglich F.
Gegenannahme: A ist nicht-prim Attribut, das transitiv bezüglich F von einem Schlüssel X für $\mathcal{S}$ abhängt. Dann gibt es ein Y mit $A \notin X \cup Y$, so dass

$$X \to Y \in F^+, \quad Y \to X \notin F^+ \quad \text{und} \quad Y \to A \in F^+ \tag{1}$$

gilt.
Nun wissen wir, dass für $Y \to A$ eine der drei Bedingungen (3NF.1), (3NF.2) oder (3NF.3) erfüllt ist. Wegen $A \notin X \cup Y$ und da A nicht-prim ist, muss es sich also um (3NF.2) handeln. Folglich ist Y ein Superschlüssel für $\mathcal{S}$.
Daraus folgt aber $Y \to X \in F^+$. Widerspruch zu (1).

Beweis von rechts nach links

Annahme: es gibt kein nicht-primes Attribut von $\mathcal{S}$, das von einem Schlüssel für $\mathcal{S}$ transitiv bezüglich F abhängig ist.
Ausserdem wählen wir eine funktionale Abhängigkeit $X \to Y \in F^+$, für die

$$Y \not\subseteq X \quad \text{und} \quad X \text{ ist kein Superschlüssel für } \mathcal{S}$$

vorausgesetzt wird. Ferner betrachten wir ein Attribut $A \in Y \setminus X$.
Wegen

$$X \to Y \in F^+$$

gilt dann auch

$$X \to A \in F^+ \quad . \tag{2}$$

Da X kein Superschlüssel für $\mathcal{S}$ ist, gibt es einen Schlüssel Z für $\mathcal{S}$ mit der Eigenschaft

$$Z \to X \in F^+ \quad \text{und} \quad X \to Z \notin F^+ \quad . \tag{3}$$

Mit (2) und (3) folgt also, dass A von einem Schlüssel für $\mathcal{S}$, nämlich Z, transitiv bezüglich F abhängig ist.
Daher ist A prim und es folgt (3NF.3).

Beispiel

Se $\mathcal{A} := (\texttt{BId}, \texttt{Name}, \texttt{Adresse}, \texttt{Datum})$ mit
$F := \big\{\{\texttt{BId}\} \to \{\texttt{Name}, \texttt{Adresse}, \texttt{Datum}\}, \{\texttt{Name}\} \to \{\texttt{Adresse}\}\big\}$.
Wir wissen:

1. `Adresse` ist ein nicht-primes Attribut,
2. `BId` ist ein Schlüssel für $\mathcal{A}$,
3. `Adresse` ist transitiv abhängig von `BId`.

Mit dem Lemma folgt, dass $\mathcal{A}$ nicht in 3NF ist.

Umgekehrt folgt aus dem Lemma auch, dass es in jedem Schema das nicht in 3NF ist, transitive Abhängigkeiten geben muss.
Somit werden in jedem Schema das nicht in 3NF ist, dieselben Anomalien auftreten, die wir am Anfang für $\mathcal{A}$ beschrieben hatten.

Lemma: Normalformen

Lemma

Es gilt folgende Beziehung:

$$\text{BCNF} \implies \text{3NF} \implies \text{2NF} \implies \text{1NF} \; .$$

Beweis: nur zeigen 3NF $\Rightarrow$ 2NF.
Sei $\mathcal{S}$ in 3NF bezüglich F, und sei A ein Attribut von $\mathcal{S}$. Ist A prim, so ist (2NF.1) erfüllt, und wir sind fertig.
Ist A nicht-prim, so folgt mit dem vorherigen Lemma:

$$A \text{ kann nicht von einem Schlüssel für } \mathcal{S} \text{ transitiv bez. } F \text{ abh. sein.} \tag{4}$$

Beweis Fortsetzung

Nun gehen wir indirekt vor und nehmen an, dass

$$\text{(2NF.2) nicht erfüllt ist.} \tag{5}$$

Das heisst, A ist von einem Schlüssel X für $\mathcal{S}$ partiell bezüglich F abhängig. Dann gibt es eine echte Teilmenge Z von X, so dass

$$Z \to A \in F^+ \tag{6}$$

gilt. Da A nicht-prim ist, kann A kein Element von X sein.
Also haben wir

$$A \notin X \quad \text{und} \quad A \notin Z \ . \tag{7}$$

Da X ein Schlüssel für $\mathcal{S}$ und Z eine echte Teilmenge von X ist, dürfen wir ferner schliessen auf

$$X \to Z \in F^+ \quad \text{und} \quad Z \to X \notin F^+ \ . \tag{8}$$

Aus (6), (7) und (8) folgt schliesslich, dass A vom Schlüssel X transitiv bezüglich F abhängig ist. Dies ist jedoch ein Widerspruch zu (4). Damit ist die Annahme (5) nicht erfüllbar und Bedingung (2NF.2) muss gelten.